```
                        title  'CP/M-86 Loader'

                ;       The CPMLDR consists of this module along with the
                ;       LDRBDOS and LBIOS.

                ;       CPMLDR resides on the first two tracks of a
                ;       CP/M-86 system diskette and is brought into memory
                ;       by the ROM bootstrap loader to load initiate CP/M-86

                ;       It opens the file 'CPM.SYS' using the LDRBDOS and LBIOS
                ;       and then reads it into memory. Finally, a jump to the BIOS
                ;       initialization entry point starts CP/M-86

                ;       The first 128 byte record of the CPM.SYS file is a header
                ;       with the following format:

                ;       ty      rb      1               ;seg type (not used here)
                ;       len     rw      1               ;length (not used here)
                ;       abs     dw      ldrstart        ;absolute segment address for LOADER
                ;       min     rw      1               ;minimum mem (not used here)
                ;       max     rw      1               ;max mem needed (not used here)

                ;       (This header record is constructed automatically by the
                ;       GENCMD utility).

                ;       CPMLDR may be read into any segment that does not
                ;       overlap the desired system load segment as it makes
                ;       all memory references using copies of the CS: register
                ;       it is entered with.

0000            false           equ     0
FFFF            true            equ     not false

000D            cr              equ     0dh
000A            lf              equ     0ah

1200            lbios_offset    equ     1200h           ; offset of LBIOS

2500            biosoff         equ     2500h           ; offset of BIOS from start of CPM.SYS
                                                        ; this is the entry point into CPM.SYS

0000            bootdrv         equ     0               ; boot drive always zero

                                                        ; 128 bytes of CPM.SYS

0406            lbdosoff        equ     406H            ;location of LBDOS in LOADER

00E0            bdos_int        equ     224             ;lbdos interrupt number

                        ; dummy section for interrupt vectors

0000                            dseg    0
```

CP/M 86 V.1.1 - vol I
COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
C81-162

```
                                        org     0
0000                    abs_zero        rw      2*bdosint
0380                    bdos_offset     rw      1
0382                    bdos_segment    rw      1

                                ; bdos function numbers

0002                    coutf           equ     2
0009                    pstrf           equ     9
000E                    seldsk          equ     14
000F                    openf           equ     15
0014                    readsf          equ     20
001A                    dmaf            equ     26
0033                    dmabf           equ     51


                        ;*********************************
                        ;*
                        ;*      CPMLDR starts here
                        ;*
                        ;*********************************

                                cseg
                                org     0                       ; JMPF to here from boot ROM

0000 E9FD11       1200          jmp     LBIOS                   ; allow loader BIOS to
                                                                ;       initialize
                        start:                                  ; loader BIOS jumps here

0003 33C08ED8                   xor ax,ax ! mov ds,ax           ; temp DS at absolute zero
0007 C70680030604               mov bdos_offset,lbdosoff        ;  to patch in interrupt table
000D 8C0E8203                   mov bdos_segment,cs             ;   offset and segment
0011 8CC88ED0                   mov ax,cs ! mov ss,ax           ; make ss, ds, es = cs
0015 8ED88EC0                   mov ds,ax ! mov es,ax
0019 BC3801                     mov sp,offset(stack)            ; set up local stack

001C E89200       00B1          call initlbdos                  ;warm up lbdos and lbios
001F E89700       00B9          call openfnc                    ;open CPM.SYS
0022 3CFF7509     002F          cmp al,255 ! jne perr           ; insure good file
0026 BAE100E8A400 00D0          mov dx,offset nofile ! call msg ; no CPM.SYS file
002C E9AE00       00DD          jmp stop                        ;  then halt the machine
                        perr:
002F 8CCAE89400   00C8          mov dx,cs ! call setdmab
0034 BA8801E83700 00C3          mov dx,offset page1 ! call setdma       ;read first page of CPM.SYS
003A E89800       00D5          call read
003D A1B[01]                    mov ax,word ptr page1+3                 ;get abs field from header
0040 2EA35501                   mov ldseg,ax                    ; save it and
0044 8BD0E87F00   00C8          mov dx,ax ! call setdmab        ; set DMA segment for disk IO
0049 BA2601E88100 00D0          mov dx,offset segment ! call msg        ; put memory map on console
004F 2EA15501E836 008C          mov ax,ldseg ! call phex                ; print base system segment
     00           008C
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
                        ;
0056 BA0000                     mov dx,0                        ;offset of CPM in segment
                        readit:
0059 E86700     00C3            call setdma                     ; set DMA offset for
005C 52E87500   00D5            push dx ! call read             ; next sector read
0060 3C017415   0079            cmp al,01H ! je done            ; check for EOF
0064 3C007409   0071            cmp al,0 ! je prerr             ; check for good write
0068 BA0B01E86200 00D0          mov dx,offset rerr ! call msg   ; print READ ERROR message
006E E76C00     00DB            jmp stop                        ; hard stop on any error

0071 5A81C28000         prerr:  pop dx ! add dx,80h             ; address for next record
0076 E9E0FF     0059            jmp readit
                        done:
0079 BA3B01E85100 00D0          mov dx,offset lenmsg ! call msg ; print length message
007F 5848E80800 008C            pop ax ! dec ax ! call phex     ; print last address
0084 E84600     00CD            call pcrlf                      ; and a crlf
0087 2EFF2E5301                 jmpf bios                       ; leap to BIOS initialization

                        ;*****************************
                        ;*
                        ;*      subroutines
                        ;*
                        ;*****************************

                        ;******
                        phex:                           ;print 4 hex characters from ax
003C B90404                     mov cx,0404h            ; 4 in both CH and CL
                        lhex:
003F D3C0                       rol ax,cl               ; rotate left 4
0091 5150                       push cx ! push ax       ; save crucial registers
0093 E80700     009D            call pnib               ; print hex nibble
0096 5859                       pop ax ! pop cx         ; restore registers
0098 FECD75F3   003F            dec ch ! jnz lhex       ; and loop four times
009C C3                         ret
                        pnib:                           ;print low nibble in AL as hex char
009D 240F3C09                   and al,0fh ! cmp al,9
00A1 7705       00A8            ja p10                  ;above 9 ?
00A3 0430                       add al,'0'              ;digit
00A5 E90200     00AA            jmp prn
00A8 0437               p10:    add al,'A'-10           ;char a-e
00AA 8AD0               prn:    mov dl,al

                        ;******
                        putchar:
00AC B102                       mov cl,coutf
00AE E97900     00DA            jmp sys_vec

                        ;******
                        initlbdos:
00B1 B10EBA0000                 mov cl,seldsk ! mov dx,bootdrv  ; select boot disk
00B6 E92100     00DA            jmp sys_vec
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER #_____

```
                        ;******
                        openfnc:
00B9 B10FBA5701                 mov cl,openf ! mov dx,offset fcb        ; fcb already initialized
00BE B10F                       mov cl,openf
00C0 E91700       00DA          jmp sys_vec

                        ;********
                        ;
                        setdma:                         ;set new dma addr in dx
00C3 B11A                       mov cl,dmaf
00C5 E91200       00DA          jmp sys_vec

                        ;********
                        ;
                        setdmab:                        ; set new dma segment base from DX
00C8 B133                       mov cl,dmabf
00CA E90D00       00DA          jmp sys_vec

                        ;******
                        ;
00CD BA5001             pcrlf:  mov dx,offset crlf      ;print carriage return, line feed

                        ;******
                        ;
                        msg:                            ;print msg starting at dx until $
00D0 B109                       mov cl,pstrf            ;print string function
00D2 E90500       00DA          jmp sys_vec

                        ;*****
                        ;
                        read:
00D5 BA5701B114                 mov dx,offset fcb ! mov cl,readsf
                        ;       jmp sys_vec

                        ;******
                        ;
                        sys_vec:
00DA CDE0                       int bdos_int
00DC C3                         ret

                        ;******
                        ;
00DD FA                 stop:   hlt                     ; hard stop 8086 for error
00DE E9FCFF       00DD          jmp     stop

                        ;*********************************
                        ;*
                        ;*      DATA AREA
                        ;*
                        ;*********************************

00E1 0D0A54686520       nofile          db      cr,lf,'The File CPM.SYS Not Found On This Disk$'
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
                        46696C652043
                        504D2E535953
                        204E6F742046
                        6F756E64204F
                        6E2054686973
                        204469736B24
010B 0D0A4572726F       rerr            db      cr,lf,'Error In Reading CPM.SYS$'
                        7220496E2052
                        656164696E67
                        2043504D2E53
                        595324
0126 0D0A5365676D       segment         db      cr,lf,'Segment Address = $'
                        656E74204164
                        647265737320
                        3D2024
013B 0D0A20202020       lenmsg          db      cr,lf,'    Last Offset = $'
                        4C617374204F
                        666673657420
                        3D2024
0150 0D0A24             crlf            db      cr,lf,'$'


                                ; vector for jmpf indirect to start CP/M

0153 00000000           bios            dd      abs_zero                ; (dummy value)

                                org offset bios                 ; overlay preceding with DW's

0153 0025               biosstart       dw      biosoff         ; first word is BIOS offset
0155                    ldseg           rw      1               ; second is segment to put CPM.SYS

0157 0043504D2020       fcb     db      0,'CPM     ','SYS',0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
                        202020535953
                        000000000000
                        000000000000
                        000000000000
                        000000

                                org     (offset $+1) and 0FFFEh ; even address for stack
0178                            rw      32
  01B8                  stack   equ offset $

                                dseg
                                org stack
01B8                    page1   rb      128


                                                ; dummy section for BIOS init label
                                org     lbios_offset
                        lbios:
```

```
266
267                                 end
268
269
270 END OF ASSEMBLY.  NUMBER OF ERRORS:  0.  USE FACTOR:  3%
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
ABSZERO          0000 V    56#  238
BDOSINT          00E0 N    49#   56  193
BDOSOFFSET       0380 V    57#   85
BDOSSEGMENT      0382 V    58#   86
BIOS             0153 V   124   238#  240
BIOSOFF          2500 N    40#  242
BIOSSTART        0153 V   242#
BOOTDRV          0000 N    43#  157
COUTF            0002 N    62#  152
CR               000D N    35#  212  220  225  229  233
CRLF             0150 V   181   233#
DMABF            0033 N    68#  176
DMAF             001A N    67#  170
DONE             0079 L   113   120#
FALSE            0000 N    32#   33
FCB              0157 V   163   192  246#
INITLBDOS        00B1 L    91   156#
LBDOSOFF         0406 N    47#   35
LBIOS            1200 L    80   264#
LBIOSOFFSET      1200 N    38#  263
LDSEG            0155 V   101   104  243#
LENMSG           013B V   121   229#
LF               000A N    36#  212  220  225  229  233
LHEX             008F L   135#  140
MSG              00D0 L    94   103  115  121  185#
NOFILE           00E1 V    94   212#
OPENF            000F N    65#  163  164
OPENFNC          00B9 L    92   162#
PIO              00A8 L   144   147#
PAGE1            0188 V    98   100  259#
PCRLF            00CD L   123   181#
PERR             002F L    93    96#
PHEX             008C L   104   122  133#
PNIB             009D L   138   142#
PRERR            0071 L   114   118#
PRN              00AA L   146   148#
PSTRF            0009 N    63#  186
PUTCHAR          00AC L   151#
READ             00D5 L    99   112  191#
READIT           0059 L   110#  119
READSF           0014 N    66#  192
RERR             010B V   115   220#
SEGMENT          0126 V   103   225#
SELDSK           000E N    64#  157
SETDMA           00C3 L    98   111  169#
SETDMAB          00C8 L    97   102  175#
STACK            01B8 N    89   255#  258
START            0003 L    82#
STOP             00DB L    95   116  203#  204
SYSVEC           00DA L   153   158  165  171  177  187  197#
TRUE             FFFF N    33#
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____